

ENTERPRISE OPERATIONS MANAGEMENT

SECURING VIRTUAL PRIVATE NETWORKS

Duncan Napier

INSIDE

PPTP; IPSec; L2TP; VPNs and the Future of Secure Communication Environments

INTRODUCTION

In many organizations, wide area networking (WAN) interconnectivity is achieved through such services as Frame Relay, ATM, or leased lines. Typically, these technologies require building private networks using dedicated lines. While these systems can be highly fault-tolerant, they are expensive to set up and maintain, and are often nonscalable and highly inflexible once fully provisioned and configured. As a result, such networks are often restricted to larger institutions or those running high-value transactions. For Frame Relay and ATM, a layer of privacy is established through the separation of various data channels with permanent virtual circuits (PVCs). The Open System Interconnect (OSI) Reference Model refers to the data-link layer as a layer 2 service. Frame Relay and ATM PVCs are examples of layer 2 services.

The proliferation of affordable, high-speed Internet access has created a widely accessible public network that is now competing head-to-head with traditional WAN solutions. The Internet is now a ubiquitous public network that spans the globe through a patchwork of fiber optic, copper, and wireless links. A large number of different types of equipment sits on the edges of these links and all seamlessly interconnect through the core of the Internet using the Internet Protocol (IP). An IP-capable device can, from any access point to the Internet, connect to any other device on the Internet that supports IP. IP is a layer 3 protocol under the OSI Reference Model.

PAYOFF IDEA

The recent explosive growth of a worldwide public network, the Internet, has pushed interconnectivity to unprecedented levels. This growth has, in turn, driven the trend toward increased data and information sharing, distributed computing, pervasive networks, and more mobile workforces. As the number and capabilities of mobile communication/networking devices increase, there will be a demand for secure, scalable, and standardized means of interconnecting a diverse universe of fixed and mobile nodes. This article examines current security options.

A publicly shared network such as the Internet suffers two key disadvantages. The first is that the universally accessible nature of the Internet makes eavesdropping, packet manipulation, and the forging of identities (spoofing) a major concern. Virtual private networks (VPNs) currently provide protection from each of these threats through encryption, data-integrity checking, and authentication. This article discusses the current state of some common VPN standards. VPN standards address data encapsulation at the source, tunneling of data through various media and protocols, and un-encapsulation at the destination. Encryption, error checking, and authentication at the end points are also addressed by these standards. Layer 2-based protocols are highly flexible and can tunnel a wide variety of network protocols, such as Microsoft's NetBEUI, Appletalk, or Novell's IPX/SPX. Layer 3-based VPNs are generally IP-based and can often be seamlessly and transparently deployed on any TCP/IP-based network, regardless of size or complexity.

The second disadvantage is that, generally speaking, there is no guaranteed quality of service (QoS) for network bandwidth, time delay, and latency for network traffic on the Internet. This is because IP is a connectionless and unreliable protocol and defers error detection and recovery to higher layers (e.g., the Transmission Control Protocol, TCP). As a result, Internet connections are vulnerable to slowdowns caused by traffic congestion. Congestion can be caused by such things as a failure or defect at a node on the Internet, sudden surges in demand for services, or malicious denial-of-service (DoS) attacks.

Various approaches are being deployed to alleviate congestion, ranging from Web caching to new QoS standards and protocols. For example, IP ToS (Type-of-service) headers can be in conjunction with such protocols as Resource Reservation Setup Protocol (RSVP) and Multi-Protocol Label Switching (MPLS). These protocols "tag" and differentiate IP packets and allow for enforcement rules and policies based on their tags. These standards/protocol-based options are not currently widely deployed on the Internet and may not be in the foreseeable future. The IP-based VPNs described in this article do not deal directly with this QoS issue but could, in principle, run transparently on top of QoS solutions such as the ones mentioned previously.

The VPN Consortium (<http://www.vpnc.org>) supports three major standards:

1. Point-to-Point Tunneling Protocol (PPTP)
2. IPSec
3. Layer 2 Tunneling Protocol (L2TP) over IPSec

It appears that IPSec is poised to become the dominant standard because it utilizes the IP and is supported in the IPv6 standard. L2TP over IPSec is a newer standard and possesses the advantage that it is not restricted to

EXHIBIT 1 — PPTP Data Structure



exclusively tunneling IP traffic between IP end points. PPTP is a proprietary protocol that is widely supported, most notably by Microsoft. The following sections provide an overview of each of these standards and discusses what the future may hold in store for them. Relevant IETF RFCs, Internet Drafts, and standards can be found at <http://www.vpnc.org/rfcs.html> and <http://www.vpnc.org/ids.html>.

PPTP

PPTP is a tunneling protocol that encapsulates the datagrams of network protocols in an IP packet. This allows PPTP to route many network protocols across an IP-based network such as the Internet. Once encapsulated, the PPTP packet is treated as an IP packet by any device that the packet traverses.

PPTP is based around Microsoft's Remote Access Services (RAS) for Windows NT. Microsoft Windows 98 and later support dialup PPTP networking, and NT Workstation supports LAN-to-LAN PPTP networking through the Windows NT Option Pack. Legacy versions of Microsoft Windows as well as MacOS are fully supported through such products as NTS Tunnel Builder (<http://www.nts.com>). Vendors such as Lucent, Nortel, and 3Com make remote access switches that support PPTP. Linux also supports PPTP.

The network protocol payload datagram is encapsulated and the GREv2 (Generic Routing Encapsulation protocol, V2; see IETF RFCs 1701 and 1702) contains information on the payload as well as client/server connection information. [Exhibit 1](#) illustrates the data structure of PPTP data. The header and payload is, in turn, encapsulated in an IP datagram. The payload datagram could contain an IPX, NetBEUI, or another IP packet. Because PPTP is layer 2 based, it also contains data-link information.

RAS connections are typically initiated through PPP (Point-to-Point Protocol) connection. PPP is an encapsulating protocol for transmitting the datagrams of various network protocols over point-to-point links. Authentication is carried out through PPP using one of the following authentication protocols: Extensible Authentication Protocol (EAP), Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP), CHAP, Shiva Password Authentication Protocol (SPAP), or Password Authentication Protocol (PAP-cleartext). The authentication token or password is authenticated either by validating a dialup-network-enabled

user account on the RAS server or through Microsoft Domains services. Note that the secureness of the connection rests solely on the strength of the user password. This contrasts with most other modern authentication procedures that rely on digital certificates or cryptographic keys for security.

Once the connection has been validated, RAS strips off the delivery, IP and GRE headers, and handles the PPP connection just as it would a regular dialup PPP connection. Packets are then transmitted through the network as local traffic. By default, PPTP clients use TCP port 1723.

PPTP attained wide usage for a variety of reasons. It was bundled free of charge with a variety of Microsoft products. It transparently uses Microsoft Networking's Domain authentication structure for validation of users. PPTP is fairly straightforward and inexpensive to implement. PPTP supports multi-protocol tunneling, which means that small companies that use NetBEUI-based networks or Novell IPX on their LANs could allow remote users to connect using dialup modems or the Internet.

It appears that Microsoft's strategy focus has moved away from PPTP and toward IPsec (or more specifically, L2TP over IPsec) with Windows 2000 and Windows XP. However, PPTP and RAS currently remain in very widespread use and will probably continue to do so for the foreseeable future.

Microsoft's implementation of PPTP (and not, notably, PPTP itself) has been publicly criticized by some noted security experts, such as Bruce Schneier of Counterpane Systems (<http://www.counterpane.com/pptp-pressrel.html>). Schneier contends that the security of Microsoft's implementation of PPTP is irredeemably broken and recommends IPsec as an alternative (<http://www.counterpane.com/pptp-faq.html>). The PPTP implementation in Windows 2000 was intended to address some of these concerns. Another criticism of PPTP is that it does not specifically address scalability issues (i.e., management of large numbers of nodes).

IPSEC

IPsec is an extension to the Internet Protocol (IP) that provides authentication and encryption at layer 3 (transport layer) of the OSI Reference Model. IPsec has been mandated into the IETF's specification of IP version 6 (IPv6); and as a result, any IPv6 implementation must support IPsec natively.

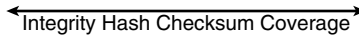
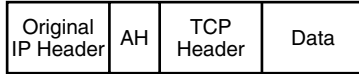
IPsec is really a collection of protocols. Three protocols are used to handle encapsulation, encryption, and authentication: AH (Authentication Header), ESP (Encapsulating Security Payload), and IKE (Internet Key Exchange). Authentication, encapsulation, and key management can be made completely transparent to the end user. IPsec gateways can be set up with no end-user action. End users need not even be aware that they are using IPsec to tunnel data over an insecure network.

EXHIBIT 2 — IPSec Authentication Header

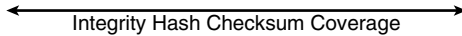
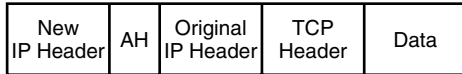
(a) Original IP Packet



(b) Transport Mode



(c) Tunneling Mode



AH and ESP handle encryption and authentication. AH is added after the IP header but before the data (payload). Refer to [Exhibit 2](#) for the data structure of IPSec data that uses AH. The AH carries authentication information at the packet level and is not involved in encryption. The AH is typically 96 bits and is a hash/digest of an authentication token (shared secret). The secret is commonly hashed using the MD5 Message Digest Algorithm or SHA (Secure Hash Algorithm). Authentication at this level with cryptographic keys and digital certificates is too slow for most practical purposes. IPSec does not specify procedures for encryption and authentication, but instead provides an open framework that allows for the implementation of industry-standard algorithms. AH is purely an authentication device used to verify that the sender is who he says he is. The AH does not carry out encryption.

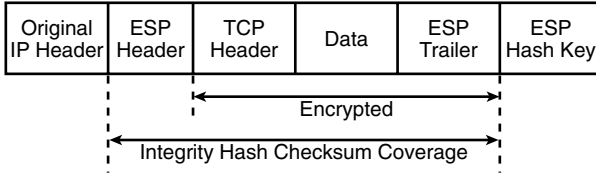
ESP provides for one or both of encryption and authentication. [Exhibit 3](#) illustrates this. It may be used in conjunction with AH or without AH. While it is possible to set up tunnels with either authentication *or* encryption deployed, it leaves communications open to numerous forms of attack. Encryption is carried out using a block cipher (a symmetric or shared-key cipher operating on fixed-size blocks of plaintext), with 3DES being commonly used. RFCs for the deployment of other encryption algorithms such as IDEA, Blowfish, and RC4 have been published and are actively supported by vendors. The encryption keys are shared using IKE.

EXHIBIT 3 — Encapsulating Security Payload

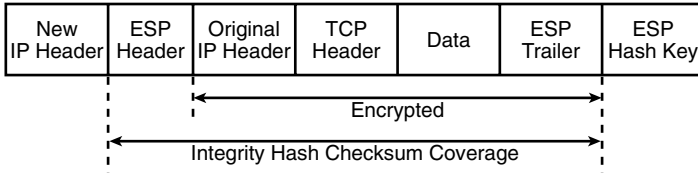
(a) Original IP Packet



(b) Transport Mode



(c) Tunneling Mode



IKE negotiates the connection parameters, including the initialization, handling, and renewal of encryption keys. Authentication is carried out using privately shared secrets (e.g., a passphrase) or cryptographic keys that guarantee the identity of the two parties. There is also an implementation for using X.509 certificates. The Diffie-Hellman method using cryptographic key-pairs is usually to exchange authenticators. Bulk encryption algorithms, such as triple DES or Data Encryption Standard are used to encrypt data. Hash algorithms such as MD5 and SHA provide authentication of each packet. IKE also negotiates the regeneration of encryption keys at specific time intervals for added security.

IKE uses UDP port 500. ESP requires that port 50 and AH requires that port 51 be open and accessible for control and communication.

IPSec can be used to support secure and transparent subnet-to-subnet, point-to-point, or point-to-subnet tunnels. IPSec has two modes: transport mode and tunnel mode. Transport mode provides source-to-destination protection of the datagrams only (Exhibits 2(b) and 3(b)). It authenticates, encapsulates, and encrypts the IP data only, but leaves the transport headers (IP information) intact. As a result, transport mode is typically used to build authenticated host-to-host encrypted tunnels. Tunnel mode creates a new IP header, encapsulating the entire original IP

datagram (Exhibits 2(c) and 3(c)). This effectively hides information about the original sender. Tunnel mode is typically used for communication with or between private subnets using NAT (network address translation).

IPSec inserts header and encrypted payload data into an existing IP (often non-IPSec) packet. This allows IPSec data to traverse all existing IP networks. A primary reason for the rapid rise of IPSec (in contrast to the next version of IP, IPv6) is that all IP networks (including IPv4) can pass full-featured IPSec traffic with no modification to the network whatsoever. IPSec has been implemented both in hardware and software. All major vendors of network hardware and software applications support IPSec networking. Virtually all current networked operating systems also support IPSec. IPSec also enjoys considerable support as in the Open Source form. It is instructive to note that while virtually all IPSec implementations are compliant with the existing RFCs, they may not necessarily be interoperable. It is recommended that IPSec implementations from different vendors be tested to ensure that they are fully compatible.

Some of the secondary issues that arise from the wide-scale deployment of encryption tunnels are ease of configuration, scalability, and compatibility.

Packet filtering and control mechanisms that rewrite packet headers often break IPSec. One prime example is NAT. This is because modifying IP packet headers break IPSec's integrity checks. Typically, this situation restricts the tunneling of IP traffic through gateways that perform header translation. As a result, un-encapsulation must occur prior to header translation. An IETF Internet Draft has been put out with some proposals regarding the interaction of IPSec and NAT.

The scalability issue arises from the fact that manual configuration of encryption tunnels can become arduous as the number of interconnected nodes grows. One proposed solution uses extensions to BIND to create a Domain Name Server (DNS)-based public key infrastructure (called Secure DNS). In this case, there is no need to explicitly configure tunnels. When a packet leaves the network interface, the DNS lookup checks for the public key associated with its IP address destination. If a public key for the destination is published, IKE negotiates a key exchange with the remote node and sends the packet fully encrypted. This process has been implemented as so-called "opportunistic encryption" and an IETF Internet Draft discusses this matter in detail. The idea of a universal DNS-based public key infrastructure (PKI) for the whole Internet is intriguing and warrants further development. IPSec is, at the time of writing, probably the dominant VPN protocol for creating scalable secure networks.

Most current implementation of IPSec are restricted to tunneling IP traffic. As a result, IPSec cannot be applied to networks that do not support IP. This issue has been addressed by Microsoft and Cisco through the IETF standard Layer 2 Tunneling Protocol, L2TP.

L2TP

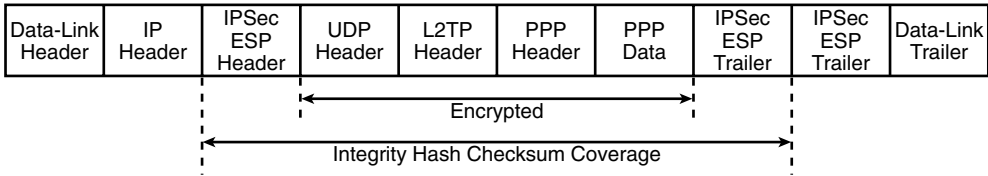
L2TP was co-developed by Microsoft and Cisco as an open standard for secure multi-protocol tunneling over IP networks. In theory, this would include protocols such as IPX, DECNet, NetBIOS, and Appletalk, but the motivation to support legacy networking protocols is often nonexistent. As a result, many current L2FP implementations (such as the native Windows 2000 one) support only IP. L2TP (RFC 2661) is based on Cisco's Layer 2 Forwarding Protocol (L2F; RFC2341) and PPTP. L2F was designed primarily for modems using public switched telephone network and for the encapsulation SLIP/PPP. The underlying foundation of L2TP is PPP (discussed earlier).

Like PPTP, L2TP is a layer 2 protocol. L2TP has much more stringent requirements for authentication, including the use of certificates, as opposed to PPTP, which relied on a user-assigned password. Unlike PPTP, L2TP is also affected by packet header modification and generally cannot traverse firewalls that carry out NAT. [Exhibit 4](#) shows the structure of L2TP data.

Just like PPTP, all L2FP communications are initiated with a PPP connection. L2P encapsulates the PPP header and payload with an L2TP header. The resulting L2TP-encapsulated packet is then encapsulated by UDP. In Microsoft's implementation, TCP-like end-to-end transmission reliability checks are carried out through Next-Received and Next-Sent fields in the L2TP messages. These perform a similar function to TCP's Acknowledgement Number and Sequence Number fields. L2TP uses UDP port 1701 as both the source and destination port.

L2TP is then typically encapsulated with IPsec's ESP and AH, followed by a second PPP encapsulation that prepares the data for transmission over the data-link layer. At its destination, the outer PPP header and trailer are stripped off, the IP header is removed, the IPsec headers and trailers are removed, and the payload is decrypted and authenticated using ESP and AH. The UDP header is then processed, leaving the inner PPP payload. This is then processed or forwarded appropriately, depending on the ruleset. Because the fundamental payload is still the PPP datagram, routing and forwarding do not necessarily require IP-based routing or forwarding resources. IPsec carries out host-level authentication with L2TP over IPsec. Once host authentication has been verified, user-level authentication schemes based on PPP such as EAP, CHAP, MS-CHAP, and PAP can be performed. Currently, Microsoft appears to be driving Windows 2000 and future versions of Windows toward native support for L2TP over IPsec. However, many vendors currently do not support this recently emerging standard. It seems likely that with the momentum of Microsoft behind it, IPsec/L2TP use will grow considerably in the future.

EXHIBIT 4 — Structure of L2TP over IPSec



VPNs AND THE FUTURE OF SECURE COMMUNICATION ENVIRONMENTS

Virtual private networks (VPNs) have many inherent advantages over other means of secure network communication. VPNs allow remote users to interact securely as local users. VPNs also work transparently at the transport or data-link layer and do not require that user applications be rewritten in order to use them. Contrast this with the case of the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols, which require rewriting of applications and adherence to specific programming APIs. The Secure Shell (ssh) is similarly hamstrung and requires the training of end users in order to be implemented effectively. All these protocols (SSL, TLS, ssh) only secure individual ports, while VPN protocols secure all data, irrespective of source or destination ports. Kerberos, the secure authentication system, requires a dedicated certificate/key infrastructure as well as modifications to application code. VPNs, on the other hand, provide end-to-end security and authentication that is normally invisible to the end user. Most secure transmission protocols use fast symmetric “bulk ciphers” to carry out fast encryption and decryption. The exchange of these secret symmetric keys is mediated by the exchange of public cryptographic key-pairs. The ever-present growth in processing speeds usually ensures very low performance loss while maintaining an acceptable level of security.

It is critical to note that while correctly implemented VPNs should generally be immune to traditional threats such as packet and password sniffing, replay attacks, as well as identity spoofing, they offer no protection against carelessness. Anyone who possesses your electronic identity, whether it is your private cryptographic key, privately shared key, or your host IP, can impersonate you. Note too that VPNs can in many cases protect against threats that the other secure communication technologies discussed above cannot (e.g., TCP hijacking and traffic analysis).

In theory, the task of collecting and loading the VPN end point and authentication information for each tunnel can be overcome with opportunistic encryption and a DNS-based PKI for the Internet. Suggested enhancements that allow for firewall traversal already exist in the Internet Draft form.

Given the rise of pervasive, networked computing and the steady growth of computing power, it seems inevitable that the VPN technologies examined in this article will continue to expand their role in electronic communications.

References

1. IETF RFCs and Internet Drafts regarding VPNs can be found at <http://www.vpnc.org/rfcs.html> and <http://www.vpnc.org/ids.html>.
2. A good working overview of IPSec, complete with full Linux source code, can be found at the official FreeS/WAN Web site <http://www.freeswan.org>.

-
3. A relatively recent and comprehensive overview of Windows 2000 VPNs can be found in *Windows 2000 Virtual Private Networking* by Thaddeus Fortenberry, ISBN 1-57870-246-1, New Riders Publishing, Indianapolis, IN, 2001.
-

Duncan Napier is the owner/operator of Napier Systems Research, an information technology and systems consultancy based in Vancouver, British Columbia. Duncan's educational background is in computer science and computational chemistry, and his company specializes in the design, configuration, and management of networks and network solutions. He can be reached by e-mail at napier@computer.org.